

Product Known Issues and Solutions Status Summary

Intel® C++ and Fortran Compilers for Linux* and Windows*

[C++ Compilers for Linux](#)

[Fortran Compilers for Linux](#)

[C++ Compilers for Windows](#)

[Fortran Compilers for Windows](#)

C++ COMPILERS FOR LINUX

Product	Version	Title	Last Update
Intel® C++ Compiler for Linux	5.0.1, 6.0 beta	Q91931/97361: When will assembler (ias) be built native on Itanium-based application?	12/6/2001
Symptom:	The assembler (ias) included with the Intel C/C++ Compilers for Linux and the Intel® Fortran Compilers for Linux is a 32-bit executable image. When will this be ported as an Itanium-based application?		
Current Status/Solution:			
This is a known issue. If you encounter this problem in your development please report the issue to Intel via https://premier.intel.com support web site and reference the issue Q97361. You will be informed when this issue has been resolved.			

Product	Version	Title	Last Update
Beta Intel C++ Compiler, Linux	5.0, 6.0 beta	Q101726 - EP compiler switch problem	12/11/2001
Symptom:	The -EP compiler switch in the Intel C++ Compiler for Linux, used to pre-process a C++ file to stdout while omitting #line directives does not currently omit the #line directives.		
Current Status/Solution:			
This issue is currently under investigation. If you encounter this problem in your development please report the issue to Intel via https://premier.intel.com support web site and reference issue Q101726. You will be informed when this issue has been resolved.			

Product		Version	Title	Last Update
Beta Intel C++ Compiler, Linux		6.0 beta	Q102110 - delete[] of null pointers sometimes fails	12/11/2001
Symptom:	delete[] of null pointers may fail for pointers to arrays of objects. Failure mode is segmentation fault on IA-64 and infinite loop on IA-32.			
Current Status/Solution:				
This is a known issue. If you encounter this problem in your development please report the issue to Intel via https://premier.intel.com support web site and reference the issue Q102110. You will be informed when this issue has been resolved.				

Product		Version	Title	Last Update
Beta Intel C++ Compiler, Linux		6.0 beta	Q102204 - Unable to vectorize large subroutines	12/13/2001
Symptom:	Unable to vectorize very large subroutines when compiling with -QxK. No vectorization messages are seen, even when the option -vec_report3 (-Qvec_report3) is set.			
Current Status/Solution:				
This problem is currently under investigation. If you encounter this problem in your development please report the issue to Intel via https://premier.intel.com support web site and reference the issue Q102204. You will be informed when this issue has been resolved.				

Product		Version	Title	Last Update
Beta Intel C++ Compiler, Linux		5.0, 6.0 beta	Q102612 - STL Vector performance	12/17/2001

Symptom:	The Intel C++ compiler for Linux experiences a performance degradation over gcc in STL Vector code. The Intel C++ compiler dynamically increases the Vector size by 50% rather than doubling it in order to reduce memory footprint. However this results in calling copy constructors twice as often as gcc.
Current Status/Solution:	
This tradeoff in performance vs. memory size is currently being investigated. If you encounter this problem in your development, please report the issue to Intel via https://premier.intel.com support web site and reference issue Q102612. We will let you know when this issue is resolved.	

Product	Version	Title	Last Update
Beta Intel C++ Compiler, Linux	6.0 beta	Q103502 - icc does not handle .S files on command line	12/10/2001
Symptom:	icc does not handle .S files on command line. icc will assemble a file with a suffix '.s', but will not assemble '.S'. gcc lays out specific semantics for .s and .S files, which our build process uses. Quoting from the gcc manpage: .s Assembler source; assemble .S Assembler source; preprocess, assemble		
Current Status/Solution:			
This problem is currently under investigation. If you encounter this problem in your development please report the issue to Intel via https://premier.intel.com support web site and reference issue Q103502. You will be informed when this issue has been resolved.			

Product	Version	Title	Last Update
Beta Intel C++ Compiler, Linux	5.0.1, 6.0 beta	Q105001:-Wl,<options> flag not supported	12/13/2001
Symptom:	The -Wl,<options> flag is not recognized by the Intel C++ Compiler and may cause the linking process to hang. The -Wl,<options> flag is the GNU gcc command to pass options to the linker.		

Current Status/Solution:

This issue is currently under investigation. If you encounter this problem in your development please report the issue to Intel via <https://premier.intel.com> support web site and reference issue Q105001 and we will inform you when this issues is resolved.

The Intel compiler syntax for passing options to the linker, -Qoption,ld,<option>. Example:
icc -Qoption,ld,-rpath -Qoption,ld,/lib:/usr/lib a.c -o a

Product	Version	Title	Last Update
Intel C++ Compiler for Linux	5.0.1, 6.0 beta	Q80444 - C++ function templates: wrong overloaded function chosen	12/5/2001

Symptom:

The following translation unit should produce an unresolved external to the non-template function overload. Instead, the function template is instantiated (and fails in this case).

```
struct S { };

void f(const S&);

// should not be instantiated, but is
template<class T>
void f(const T&) { T::oops(); }

int main()
{
  S s;
  f(s);
}
```

Current Status/Solution:

This problem is currently under investigation. If you encounter this problem in your development please report the issue to Intel via <https://premier.intel.com> support web site and reference issue Q80444. You will be informed when this issue has been resolved.

Product	Version	Title	Last Update
Intel C++ Compiler for Linux	6.0 beta	Q86141 - (expr)? memberfunc() : memberfunc(); refused by Intel compiler	12/7/2001

Symptom:	<p>To demonstrate the problem, compile the attached example as: ecpc -c ambop=.cxx and note that the use of "cerr" works OK, and the next line doesn't.</p> <p>The Intel Linux/64 compiler cannot disambiguate (i.e. correctly choose) a member function when it appears in the context of a logical no-op in a ?: statement. The context is: cout<< (expression)? flush : flush; That is, regardless of the (expression), the flush iomanipulator is called. By contrast, the statement: cout<< flush; has no problem. Additionally, other C++ compilers (and their runtime system as declared by accompanying headers) have no problem with such statements - including gcc and ecl.</p>
Current Status/Solution:	
<p>This problem is currently under investigation. If you encounter this problem in your development please report the issue to Intel via https://premier.intel.com support web site and reference issue Q86141. You will be informed when this issue has been resolved.</p>	

Product	Version	Title	Last Update
Intel C++ Compiler for Linux	5.0.1, 6.0 beta	Q104682:Getting undefined reference to `_alloca_probe' error when using alloca() function	12/6/2001
Symptom:	<p>Attempts to use the alloca() function from within a shared library give a link error when linking the programs using this shared library:</p> <pre> \$ cat > x.c <<EOF #include <alloca.h> void call_alloca (int arg) { alloca (arg); } EOF \$ cat > y.c <<EOF int main (int argc, char *argv[]) { call_alloca (argc); return 0; } EOF \$ icc -g -O2 x.c -shared -o libx.so x.c: \$ icc y.c libx.so y.c: libx.so: undefined reference to `_alloca_probe'</pre>		

Current Status/Solution:

This problem is currently under investigation. If you encounter this problem in your development please report the issue to Intel via <https://premier.intel.com> support web site and reference issue Q104682. We will inform you when the issue is resolved.

Product	Version	Title	Last Update
Intel C++ Compiler, Linux	5.0.1, 6.0 beta	Q97293 - Passing multiple arguments to Linux linker results in ld: unrecognized option	12/13/2001
Symptom:	When I try to pass an option that takes an argument to the linker (ld) , I get an error: ld: unrecognized option For example: <code>icc hello.c -Qoption,ld,"-Map hello.map"</code> produces the error: <code>ld: unrecognized -a option `p hello.map'</code>		

Current Status/Solution:

To pass an option that takes an argument to the linker, the option and its argument should be separated by commas, and no white space should appear between the two: `icc hello.c -Qoption,ld,-Map,hello.map`

Optionally, the option and argument can be quoted, again with no white space: `icc hello.c -Qoption,ld,"-Map,hello.map"`

This is a known issue. If you encounter this problem in your development please report the issue to Intel via <https://premier.intel.com> support web site and reference issue Q97293. We will notify you when the issue is resolved.

FORTTRAN COMPILERS FOR LINUX

Product	Version	Title	Last Update
Beta Intel Fortran Compiler, Linux	6.0 beta	Q106051 ld: Warning: type of symbol `__libc_start_main' changed from 1 to 2 in libc-start.o	12/7/2001
Symptom:	At link time, you may see the following warning: ld: Warning: type of symbol `__libc_start_main' changed from 1 to 2 in libc-start.o		

Current Status/Solution:

This problem is currently under investigation. If you encounter this problem in your development please report the issue to Intel via <https://premier.intel.com> support web site and reference issue Q106051. You will be informed when this issue has been resolved.

Product	Version	Title	Last Update
Intel Fortran Compiler for Linux	5.0.1, 6.0 beta	Q96097: Can't build shared library using Linux compilers	12/17/2001
Symptom:	When I build using the Intel C/C++ Compilers for Linux and the Intel Fortran Compilers for Linux, I get the following errors: ld: bfd assertion fail elf64-ia64.c:2991 ld: bfd assertion fail elf64-ia64.c:2991 ld: bfd assertion fail elf64-ia64.c:2991 ld: bfd assertion fail elf64-ia64.c:2991 ld: bfd assertion fail elf64-ia64.c:2991 efc: error: ld: core dumped efc: error: Fatal error in ld, terminated by unknown signal(139) How do we build executables that use shared libraries?		
Current Status/Solution: Verify that the config file does not contain -static, and use -i_dynamic in the build command. The ifc.cfg and icc.cfg files are the configuration files for the compilers for 32-bit applications. They are found in the /opt/intel/compiler50/ia32/bin directory. The efc.cfg and ecc.cfg files are the configuration files for the compilers for Itanium™-based applications. They are found in the /opt/intel/compiler50/ia64/bin directory. The -i_dynamic switch ensures that the Intel libraries will be linked in dynamically. This is a known issue. If you encounter this problem in your development please report the issue to Intel via https://premier.intel.com support web site and reference the issue Q96097. You will be informed when this issue has been resolved.			

C++ COMPILERS FOR WINDOWS

Product	Version	Title	Last Update
Beta Intel C++ Compiler, Windows	6.0 beta	Q103075:Compiler Selection Tool broken after updated install	12/17/2001
Symptom:	<p>Q103075: I uninstalled old 5.0 version of the Intel C++ compiler for Windows and installed the latest beta version (6.0-31). While trying to select the Select Compiler tool from the Tools menu in Microsoft Visual Studio*, I get the following message</p> <pre>"Error: get_compiler_choice_info return code=1"</pre>		

Current Status/Solution:

Edit register entries using regedit.

Go into HKEY_CURRENT_USER->Software->Intel->Intel Tools->Select Compiler, Here edit the value for Compiler_List and remove the number starting with 50..., similarly for Compiler_list_64.

For example
I had compiler_list = 50032, 60032
I changed it to compiler_list = 60032.

In the example shown, this is an error in the uninstall script of the icl 5.0 compiler, which should have made this change.

This is fixed in icl 6.0 Beta update compiler. (build date after 12/1/2001).

Product	Version	Title	Last Update
Intel C++ Compiler for Windows	5.0.1, 6.0 beta	Q83144 - printf invalid format string conversion: warning #269	11/30/2001
Symptom:	<div>There are cases where the compiler reports invalid format string conversion in printf/fprintf/etc. functions when they are valid. This happens when using "%S" to specify wide character strings when using the C runtime library's printf (also "%ws", "%wS", "%ls" and "%lS") or to specify single character strings when using the C runtime library's wprintf (also "%hs" and "%hS").</div> <div>#include <stdio.h></div> <div>int main(int argc, char *argv[])</div> <div>{</div> <div>wchar_t widestring[] = L"A Wide String";</div> <div>char narrowstring[] = "A Narrow String";</div> <div></div> <div>char widedest[100];</div> <div>char narrowdest[100];</div> <div></div> <div>// Convert a wide string to a narrow string - 5 different ways</div> <div>sprintf(narrowdest, "%S", widestring); // because we are a non-wide char app, "%S" means a wide string.</div> <div></div> <div> // If we were a wide char app, "%S" would mean a normal char string</div> <div>sprintf(narrowdest, "%ls", widestring);</div> <div>sprintf(narrowdest, "%lS", widestring);</div> <div>sprintf(narrowdest, "%ws", widestring);</div> <div>sprintf(narrowdest, "%wS", widestring);</div> <div></div> <div>// Convert a narrow string to a wide string - 2 different ways</div> <div>sprintf(widedest, "%hs", narrowstring);</div> <div>sprintf(widedest, "%hS", narrowstring);</div> <div></div> <div>return 0;</div> <div>}</div>		

Current Status/Solution:

This problem is currently under investigation. If you encounter this problem in your development please report the issue to Intel via <https://premier.intel.com> support web site and reference issue Q83144. We will inform you when this issue has been resolved.

Product	Version	Title	Last Update
Intel C++ Compiler for Windows	6.0 beta	Q87681:Cannot compile C++ program using the Intel int _bswap(int) intrinsic function	12/13/2001
Symptom:	Compiling a C++ program which calls the Intel-defined intrinsic _bswap(int) produces the error: identifier "_bswap" is undefined A C program which use this intrinsic compiles, links, and runs correctly.		

Current Status/Solution:

This is a known issue. If you encounter this problem in your development please report the issue to Intel via <https://premier.intel.com> support web site and reference issue Q87681. We will notify you when the issue is resolved.

Product	Version	Title	Last Update
Intel C++ Compiler for Windows	6.0 beta	Q95542: fatal error LNK1169: one or more multiply defined symbols found building DLL or application	12/11/2001
Symptom:	Building a DLL or application that includes a header file in two or more source files, and the header file defines a function inside an unnamed namespace . Linking fails with the error message: "fatal error LNK1169: one or more multiply defined symbols found"		

Current Status/Solution:

This is a known issue. If you encounter this problem in your development please report the issue to Intel via <https://premier.intel.com> support web site and reference issue Q95542. We will notify you when the issue is resolved.

Product	Version	Title	Last Update
Intel C++ Compiler for Windows	5.0.1, 6.0 beta	Q104741:Internal Compiler Error when using the /GX compiler option	12/12/2001
Symptom:	The compiler may report the following error when you use the /GX option while compiling a C++ program containing exception handling code: "C/C++: Internal compiler error: please report to Intel (IP=0x005cf604, addr=0x0547be54) compilation aborted for icpi.narrowed.hole.cpp (code 1) " The IP address may differ depending on the program operation.		
Current Status/Solution:			
This problem is currently under investigation. If you encounter this problem in your development please report the issue to Intel via https://premier.intel.com support web site and reference issue Q104741. We will inform you when the issue is resolved.			

Product	Version	Title	Last Update
Intel C++ Compiler for Windows	5.0.1	Q105681: IPO warnings (unknown error) given even when not using IPO	12/13/2001
Symptom:	The following warning appears even when not specifying -Qipo: IPO compilation(228): (col. 8) warning: unknown error		
Current Status/Solution: This problem is currently under investigation. If you encounter this problem in your development please report the issue to Intel via https://premier.intel.com support web site and reference issue Q105681. You will be informed when this issue has been resolved.			

FORTRAN COMPILERS FOR WINDOWS

Product	Version	Title	Last Update
Intel Fortran Compiler for Windows	5.0.1, 6.0 beta	Q103496:Install cannot load web page using Netscape Navigator	12/13/2001

Symptom:	If your default web browser is the Netscape Navigator, the compiler installation is unable to load a web page from your disk that contains the link to install the compiler. As a result, the installation will not successfully complete.
Current Status/Solution:	
<p>This problem is under investigation. If you encounter this problem in your development please report the issue to Intel via https://premier.intel.com support web site and reference issue Q103496 and we will inform you when this issues is resolved.</p> <p>To workaround this issue go to the temporary directory where the compiler was extracted and manually run setup.exe.</p>	

Intel, the Intel logo, and Itanium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Copyright © 2001 Intel Corporation.

*Other names and brands may be claimed as the property of others.